Amendments to the Claims:

A listing of the entire set of pending claims (including amendments to the claims, if any) is submitted herewith per 37 CFR 1.121. This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1. A method for training a self ordering map for use in a computing system, comprising ~~the steps of~~:

initializing a set of weights of ~~a~~ the self[[-]]ordering map; and

iteratively training ~~said~~the weights over many training epochs;

wherein

for at least a number of ~~said~~the training epochs, ~~said step of~~ iteratively training ~~including~~ the weights includes

updating ~~said~~the weights based on a learning rate that is generated according to a function that changes in a fashion that is other than monotonically a decreasing ~~value~~ with the training epochs.

2. A method as in claim 1, wherein ~~said step of iteratively training includes updating said weights based on a learning rate that is generated according to a~~ the function includes a random or pseudorandom function.

3. A method as in claim 2 wherein ~~said step of iteratively training includes updating said weights based on a learning rate that is generated according to a function that is such that values over which said learning rate may~~ the random or pseudorandom function has a range that decreases with the training epochs.

4. A method as in claim 2 wherein ~~said step of iteratively training includes updating said weights based on a learning rate that is generated according to a function that is such that values over which said~~

the random or pseudorandom function is configured such that the learning rate tends to decrease with the training epochs.

5. A method as in claim 1 wherein ~~said step of iteratively training includes updating said weights based on a learning rate that is generated according to a function that is such that values over which said learning rate may~~

the function has a range that decreases with the training epochs.

6. A method as in claim 5 wherein ~~said step of iteratively training includes updating said weights based on a learning rate that is generated according to a function that is such that values over which said~~

the function is configured such that the learning rate tends to decrease with the training epochs.

7. A method as in claim 1 wherein ~~said step of iteratively training includes updating said weights based on a learning rate that is generated according to a function that is such that values over which said~~

the function is configured such that the learning rate tends to decrease with the training epochs.

8. A method of training a self ordering feature map for use in a computing system, comprising ~~the steps of~~:

choosing a random value for initial weight vectors;

drawing a sample from a set of training sample vectors and applying it to input nodes of ~~said~~the self ordering feature map;

identifying a winning competition node of ~~said~~the self ordering feature map according to a least distance criterion;

adjusting a synaptic weight of at least ~~said~~the winning node, using;

~~said step of adjusting including selecting a value for~~ a learning rate ~~used~~ to update ~~said~~the synaptic weight that is based on a function other than one that is monotonic with subsequent training epochs;

iteratively repeating ~~said steps of~~ the drawing, identifying, and adjusting to form each subsequent training epoch.

9. A method as in claim 8, wherein ~~said step of adjusting includes updating said weights based on a learning rate that is generated according to~~

the function corresponds to a random or pseudorandom function.

10. A method as in claim 9 wherein ~~said step of adjusting includes updating said weights based on a learning rate that is generated according to a function that is such that values over which said learning rate may~~

the function has a range that decreases with subsequent training epochs.

11. A method as in claim 9 wherein ~~said step of adjusting includes updating said weights based on a learning rate that is generated according to a function that is such that values over which said~~

the function is configured such that the learning rate tends to decrease with subsequent training epochs.

BEST AVAILABLE COPY

12. A method as in claim 8 wherein ~~said step of adjusting includes updating said weights based on a learning rate that is generated according to a function that is such that values over which said learning rate may~~

the function has a range that decreases with subsequent training epochs.

13. A method as in claim 12 wherein ~~said step of adjusting includes updating said weights based on a learning rate that is generated according to a function that is such that values over which said~~

the function is configured such that the learning rate tends to decrease with subsequent training epochs.

14. A method as in claim 8 wherein ~~said step of adjusting includes updating said weights based on a learning rate that is generated according to a function that is such that values over which said~~

the function is configured such that the learning rate tends to decrease with subsequent training epochs.